Research Report 231

A Comparison of Filters For Audio Signal Segmentation in Audio Restoration

Hugh Scott, Roland Wilson

RR231

This report is concerned with the choice of filters for the estimation and detection from a time-frequency representation of audio signals of partials - the approximately sinusoidal components, typically generated in harmonic series, of which many musical signals are composed. The report contains a detailed theoretical and experimental comparison between filters with Gaussian and bilateral exponential time envelopes, in terms of performance on noisy data and simplicity of implementation. It is concluded that the exponential filter is better on both counts. There follows a brief examination of the problem of onset detection, in which both additive and multiplicative approaches are examined. Some tentative conclusions are drawn about the most effective way to solve the problem of partial detection in noisy signals.

Department of Computer Science
University of Warwick
Coventry CV4 7AL
United Kingdom

October 1992

Contents

List of Figures

Introduction

In order to perform audio restoration on a perceived musical signal, where the perceived signal $x(t)$ can be considered to be made up of signal $s(t)$ and noise $n(t)$:

$$x(t) = s(t) + n(t)$$

knowledge of how the signal $s(t)$ behaves is required in order to segment it as well as possible leaving behind the noise $n(t)$. Clearly, the more information known about how the signal $s(t)$ behaves, the better estimate of $s(t)$ given $x(t)$ will be.

To estimate $s(t)$ from $x(t)$, audio signal segmentation is used. Here a tradeoff is required between how precise a model is used (and thus how accurately $s(t)$ is estimated) with the generality of the segmentation and in turn of the restoration technique. For example in order to gain better accuracy in specific cases models may be used restricting the signal detection to : one instrument playing, instruments playing monophonically, for a given number of instruments knowledge of the position in time and frequency of each note or a restriction of the frequency range of the signal.

To avoid this loss of generality the method proposed for audio restoration utilises methods by Pearson in conjunction with the Multiresolution Fourier Transform (MFT) - which is shown to be the most general transform for audio signal segmentation. This method is not restricted to a particular instrument, a particular frequency range or the number of notes being played.

As each piece has a large number of sample points this conceivably involves a lot of detection, processing and therefore time. Thus the amount of computation involved with the implementation of any algorithm must be minimal.

The Pearson method of detection uses an ordinary FIR Gaussian filter to smooth for detection in time, along with a simple frequency filter derived from the detection model (partial model) used and the MFT. The time derivative of this is used for feature (partial) onset detection and is processed independently of the partial detection. Proposed here is a simpler filter $\exp[\frac{-|t|}{\alpha}]$, which is easily implementable (indeed it can be implemented recursively) and reduces the processing time.

The proposed filter $\exp[\frac{-|t|}{\alpha}]$ is analysed and compared with the Gaussian filter used by Pearson $\exp[-\frac{t^2}{\beta^2}]$ in different amounts of noise. Two methods of feature onset detection are considered, both of which use the feature detection filters thus avoiding the need to implement a further filter for this purpose, which saves both time and computation.

In section 1 there is a literature review of methods pertinent to the chosen method of segmentation, the MFT is briefly described and the signal model is discussed.In Section 2 requirements for a filter for audio restoration are discussed and the candidate filter is chosen. In section 3 a performance analysis is given of the $\exp[\frac{-|t|}{\alpha}]$ and the $\exp[-\frac{t^2}{\beta^2}]$ filters having equivalent energy concentration for given bandwidth. Section 4 contains a description of

the experiments, with results and conclusions presented in section 5 with further work in section 6.

Acknowledgements

I would like to thank Central Research Laboratories Ltd. and SERC for their generous support in funding this work.

1 Audio Signal Segmentation

Segmentation of a signal involves the detection of features $p_i : i \in N$ and estimation of a given set of parameters ϕ_{p_i} [21] where these parameters can be start time, duration and frequency, in the case of audio. As outlined in the introduction it is necessary for the purposes of this paper to consider a level of segmentation (number of parameters associated with each p_i) high enough to contain all musical features but low enough to be of as general a nature as possible. The musical features considered here are partials.

1.1 What is a Partial?

A *partial* of a given instrument playing a given note at a given time is defined as a multiple (not necessarily integral) of the instrument's note's fundamental frequency, i.e if an instrument had for a note (Θ) fundamental frequency F_Θ (Hz), then all partials of that fundamental would be of the form $\alpha F_\Theta : \alpha \in R^+$. If $\alpha \in N^+$ then αF_Θ is a *harmonic*.It is the spread of energy across these partials which gives an instrument its *timbre*.

1.2 Partial Detection - A Literature Review

Considered here are analysis/synthesis and musical transcription methods. Both require some amount of segmentation. The cases reviewed here are those of partial detection.

1.2.1 Analysis/Synthesis Techniques

Gabor stated that sound was not a purely frequency phenomenon [6], indeed it is natural for us to have a perception of both time and frequency. Work by Roads [16] exploits this with his granular synthesis technique for musical composition, in which a number of 'grains' can be turned on or off at a given time or frequency to give textured sound.

For most analysis/synthesis techniques, however, the musical signal is considered to be made up of quasi-sinusoids added together and overlapped ([7] George,Smith, Serra [18], [19]). George and Smith [7] made no allowance for inharmonic sounds [8], both harmonic and inharmonic sounds were considered to be sinusoidal: any inharmonic sounds are treated as if they were harmonic. Serra, however, used a deterministic model to segment out the partials which were explicitly detected and defined the 'residue' to be what is left over in the signal after the partials were extracted. The model used is:

$$s(t) = \sum_{r=1}^{R} A_r(t) cos[\theta_r(t)] + e(t)$$

R is number of partials
A_r is amplitude of the r^{th} partial
θ_r is the partial's phase (related linearly to the partial's frequency)
$e(t)$ is the residue

Since the Short Time Fourier Transform (STFT) is used by Serra, this approach requires some *a priori* knowledge of the signals' feature parameters in order to determine the window type (Hamming,Hanning,Kaiser,Blackmann or Blackmann-Harris) and the time win-

dow size. These choices are crucial to the detection : the higher the frequency of the partials in a piece the wider the required window for better frequency resolution. These decisions are made independently of the algorithm and affect the results quite seriously. The peaks of the partials were detected, then followed through time using 'frequency guides' given by the user. In PARSHL[19], an application of the above, Serra used the Phase Vocoder for peak detection and frequency calculation.

1.2.2 Musical Transcription Methods

Moorer [11] considered a purely monophonic signal. There were various rules imposed on the pieces of music that could be transcribed: there were a maximum of two 'voices' present and the fundamental of one note will not overlay the partial of another. These obviously make detection and therefore transcription easier, by reducing the number of conflicting signals. The method employed by Moorer was to implement a basic comb filter in the time domain of the form

$Y_n = X_n - X_{n-m}$
X_n is the n^{th} sample of the output waveform
Y_n is the n^{th} sample of the input waveform
This comb filter has magnitude-frequency response $\sqrt{\sin^2(m\omega h) + (1 - \cos(m\omega h))^2}$
h is the number of the harmonic p_h, $h \in N^+$ (see section 1.1).

By minimising $\sum_{i=0}^{N-1} \mid X_{n+i} - X_{n+i-1} \mid$ peak values of fundamental ω are found. These values of ω and harmonics of ω are bandpass filtered using Chebychev or Butterworth windows [15] in order to isolate them and then passed to a comb pitch detector for amplitude and frequency contours. The partials are segmented by thresholding so that 90% of their energy is transmitted.

Chafe, Jaffe et al.[3] took things a step forward by considering polyphonic sounds, but restricted there study to the piano. They used knowledge of the piano (see for example Vos, Pizcalzci [8], [5]) to help in their detection of partials. The transform used was a modified version of the Q Transform, The Bounded Q Transform [10],[3],[2], which effectively conjoined scale to frequency so as to allow the easy segmentation of two piano notes in the upper half octave of the piano signal being segmented. Giving frequency resolution $\Delta F \approx$ 1.635 Hz for a signal in the lowest octave and $\Delta F \approx 66$ Hz in the highest octave. Event detection starts with finding a point where there is a sufficiently large gradient for onset. Moving linear regression of length L and with starting point k in time is used to fit a ramp to the input signal by minimising the squared error:

$$E = \sum_{n=k}^{k+L-1} (y(n) - m_k n - b_k)^2$$

where $y(n)$ is input signal, k is sampling point and m_k and b_k are the regression coefficients.

This gives a starting point in time (k) for partial detection. The Bounded Q Transform domain gives information on the frequency and amplitude. Knowledge of the rhythm of the music is used to add a 'goodness' or weight to partials detected. To avoid confusion between one note's partial and others, an energy model of piano harmonics (even have more energy and odd have less) is used. This in turn helps to determine the fundamental frequency.

Watson [20] used cepstral analysis to try to keep the type of music analysed as general as possible - the only constraint being that all tones must be harmonic. The cepstrum is used to define quefrency peaks, which correspond to equidistant frequency peaks in the log power spectrum. Using frequency ratio analysis and peak detection candidate harmonics are found. The next stage is to use the Harmonic Summing Algorithm

$$H_h(f) = \sum_{i=1}^{h} 10 \log_{10} S(if)$$

where:
f frequency
h number of harmonics
$S(f)$ instantaneous power spectra

which is maximised when the first h harmonics are present. Various heuristic methods are applied which give a weighting to all estimates. These methods are based on musical rules concerning rhythm and tonal seperation for example. Partials with the same fundamental are grouped together and enter the transcription process.

Pearson used the MFT to 'get round' the uncertainty principle [23], [9], by avoiding the limitations adherent in the STFT: the MFT is effectively a stack of STFT's with varying window size. By differencing the MFT coefficients across time and filtering them with a Gaussian filter and a simple frequency filter (see sections 2.1 and [13]) partials were detected according to the partial model. Their onset was detected using a differentiated form of the partial detection window. Then the music was transcribed using partial tracking.

1.3 The Multiresolution Fourier Transform (MFT)

There follows a brief description of the MFT, and how it is applied to audio signal segmentation, in this instance. For a fuller description see Wilson et al.,Calway [21], [1].

1.3.1 The Continuous MFT

The continuous form of the MFT is

$$\hat{x}(t, \omega, \sigma) = \int_{-\infty}^{\infty} x(\chi) w(\sigma(\chi - t)) \exp[-j\omega\chi] d\chi$$

$x(t)$ is a signal, σ is a scale factor, ω is a frequency, t is time and $w(t)$ is the window function. The constraints on the window function are that it must have finite energy

$$\int_{-\infty}^{\infty} w^2(\eta) d\eta < \infty$$

the window and it's Fourier transform must be C^2

$$\frac{\partial^2}{\partial \eta^2}(w(\eta)) \quad \text{and} \quad \frac{\partial^2}{\partial \eta^2}(\hat{w}(\eta)) \quad \text{are continuous}$$

and it must be even

$$w(t) = w(-t)$$

This leaves the choice of window general. The class of window used in this work belongs to Finite Prolate Spheroidal Sequence (FPSS) [22] class. These windows maximise energy concentration in both time and frequency. The implementation is discussed further in section 1.3.2. It is easy to see that the MFT has 3 degrees of freedom: time, frequency and scale.

1.3.2 The Discrete MFT

The Discrete form of the MFT is

$$\hat{x}(i,j,n) = \sum_{k=0}^{N-1} w_n(t_k - t_i(n))x(t_k)\exp[-jt_k\omega_j(n)]$$

N is the total number of sampling points
n is the scale index
i is the time index
j is the frequency index

where N is subject to the condition $N \geq N_t(n)N_\omega(n)$ [21], $N_t(n)$ and $N_\omega(n)$ are the number of temporal and frequency sampling points respectively. Taking $N = 2^M$ gives a suitable choice for the temporal and frequency sampling intervals $\Xi(n)$ and $\Omega(n)$

$$\Xi(n) = 2^{M-1-n} \quad \text{and} \quad \Omega(n) = 2^{m-n}\pi$$

These are the sampling intervals for 100% oversampling in time. For these intervals definitions of $N_t(n)$ and $N_\omega(n)$ are

$$N_t(n) = 2^{n+1} \quad \text{and} \quad N_\omega(n) = 2^{M-n}$$

It is straightforward to see that $n = 0$ is equivalent to the oversampled DFT of $x(t)$ and $n = M$ is the original signal (see figure 1). A scale constant of 2 makes the MFT easier to implement (via DFT and FFT) and maximises the consistency across scale (see [21] and [17]). Oversampling of 100% in time is used because it makes the transform easier to invert and allows the set of basis vectors of each level of the MFT to construct a frame [4] without the need of strict linear independence [21], [4].

A relaxed FPSS is used [13] because of the oversampling. This means that the time coefficients of the window are allowed to 'spread out' by a factor (in this case 2). This allows the largest sidelobe peak outside the interval $\Xi(n)$ to be -26.3dB [21]. It should be noted

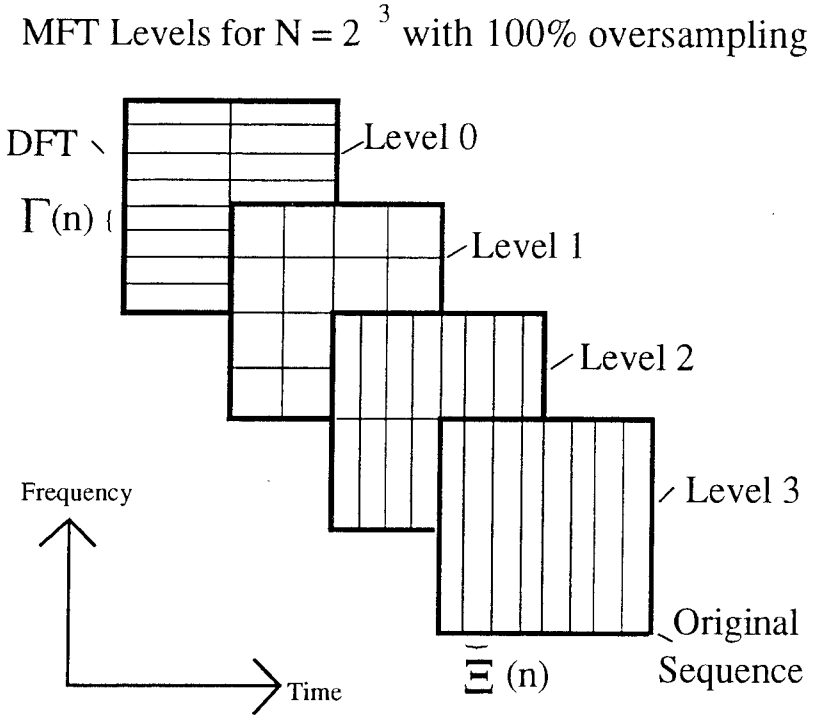


Figure 1: Time frequency plane tessellation with discrete MFT

that in this implementation the FPSS is truncated in the time domain for computational simplicity.

The MFT for an oversample of 100% can be written

$$\hat{x}(i,j,n) = \sum_{k=0}^{N-1} w_n(t_k - \frac{1}{2}\Xi(n))x(t_k)\exp[-jk\Omega(n))$$

where w_n satisfies

$$T(2\Xi(n))B(\Omega(n))T(\Xi(n))w_n = \lambda_0 w_n$$

T_{ij} is truncation operator and $B_{ij} = F_{ij}^* T_{ij} F_{ij}$ where F_{ij} is the DFT operator.

1.4 The MFT and Partial Detection

The MFT has three degrees of freedom, time (i), frequency (j) and scale (n). An arbitrary choice of frequency and time bin can be made by varying the scale parameter, allowing partials arbitrarily close in time or frequency (see [21]) to be segmented. Both the STFT and the Wavelet Transform (WT) only have two degrees of freedom. In the case of the STFT these are window size and time, and for the WT time and a scale/frequency parameter. The MFT is the most general transform for audio signal segmentation [21],[13].

1.5 The Partial Model and Wiener Filtering

In order to smooth the MFT coefficients for optimal partial detection whilst still using a general and easily implementable filter, a Wiener type filter is used.

1.5.1 Wiener Filtering

For observed signal $x(t)$ and signal $s(t)$ with noise component $n(t)$ such that :

$$x(t) = s(t) + n(t)$$

the optimal filter $h(t)$ is a solution to the Wiener-Hopf equation [12]:

$$R_{sx}(\tau) = \int_{-\infty}^{\infty} R_{xx}(\tau - \alpha)h(\alpha)d\alpha$$

where $R_{sx}(t)$ denotes the correlation of x and s. Solving this in the Fourier domain gives

$$H(\omega) = \frac{S_{sx}(\omega)}{S_{xx}(\omega)}$$

where $S_{sx}(\omega)$ denotes the transform of $R_{sx}(t)$.

Therefore in order to construct an optimal filter knowledge of the signal $s(t)$ is required, in other words a signal model is needed.

1.5.2 The Partial Model

The model used for partials is that used by Pearson [13]. Consider partial $p(t)$ with

$$p(t) = s(t) + o(t)$$

where $s(t)$ is steady state portion and $o(t)$ is onset.
where $s(t) = \Re(v(t))$ with

$$v(t) = a_{s(t)} \exp[j[t\omega_0 + \phi_v]]$$

By assuming that the partial is locally sinusoidal (see Pearson [13]), the discrete version of the MFT can be applied, showing that neighbouring frequency bins should have a phase difference of π when a partial is present. Consider a partial whose frequency ω_i lies in frequency bin j. Then because the input signal was oversampled by 100% the neighbouring frequency bin $j \pm 1$ will have a component of ω_i that is out of phase by π. Thus an indication that partials are present is when neighbouring frequency bins have the same phase after the frequency window $h(l)$ (see section 2.1) has been applied. It is shown in [13] that MFT coefficients phase-differenced in time have constant phase i.e

$$\angle \hat{x}(i,j,n)\hat{x}^*(i-1,j,n) = \omega_i - \omega_j$$

where ω_j is the centre frequency of frequency bin j.

2 Choice of Filter for Partial Detection

This section is a continuation of work by Pearson [13] on partial detection. There follows an outline of attributes required for a filter to detect partials in an audio restoration framework, a candidate filter for that purpose.

2.1 Audio Restoration Requirements for Filter

The method proposed for restoration work involves filtering a large number of MFT coefficients. For example for a one minute section sampled at $48kHz$, with an MFT level of say 10, or 12 (giving frequency resolutions of $\Delta F_{Level10} \approx 46Hz$ and $\Delta F_{Level12} \approx 12Hz$) there will be about $10,000$ coefficients in time and about 1000 coefficients in frequency. So for just three levels of the MFT (levels 10 - 12) there will be about 10^7 coefficients to filter. Obviously this takes a large amount of time, since each coefficient is filtered [12] more than once. Any filter would need to be implementable with computational efficiency.

The choice of filter in frequency $h(l)$ is the same as that for Pearson, (see sections 1.5.2 and 1.5.1), that is

$$h(l) = \begin{cases} 1 & l = 0 \\ -1 & l = 1 \\ 0 & \text{else} \end{cases}$$

For time filter $f(t)$, and MFT coefficients $\hat{x}(i,j,n)$ and frequency filter as above, the filtered MFT coefficient is written :

$$\hat{y}(i,j,n) = \sum_{l,m} h(j-l)f(i-m)\hat{x}(i,j,n)$$

with filters in both time and frequency.

2.2 Candidate Filter

One filter which fulfills all of the requirements in section 2.1 is

$$f_b(t) = \exp[\frac{-|t|}{\alpha}]$$

This bilateral exponential can be split into causal $f_c(t)$ and anti-causal $f_a(t)$ components.

$$f_c(t) = \exp[\frac{t}{\alpha}] \quad t < 0 \qquad f_a(t) = \exp[\frac{-t}{\alpha}] \quad t \geq 0$$

These can be implemented as first order recursive filters in the negative and positive time direction, allowing an increase in accuracy of filter response without significant numerical cost. This is achieved by removing the rectangular window that is imposed on non-recursive filters that are truncated in time domain and the subsequent rippling of type $\sin^2(\omega)/\omega$ in the frequency domain. The filter $f_b(t)$ has representation in the frequency domain

$$\hat{f}_b(\omega) = \frac{1}{\alpha\pi}\frac{1}{\frac{1}{\alpha^2}+\omega^2} \text{where } \omega \text{ is frequency}$$

It is obvious to see that both the $f_b(t)$ and $\hat{f}_b(\omega)$ are centered at $t = 0$ and $\omega = 0$ respectively.

The Gaussian 'equivalent' of these filters, as used by Pearson [13] is

$$g_b(t) = \exp[-\frac{t^2}{\beta^2}]$$

and for causal and anti-causal

$$g_c(t) = \exp[-\frac{t^2}{\beta^2}] \quad t < 0 \quad g_a(t) = \exp[-\frac{t^2}{\beta^2}] \quad t \geq 0$$

where $g_b(t)$, $g_c(t)$ and $g_a(t)$ will be referred to as the Gaussian equivalents of $f_b(t)$ $f_c(t)$ and $f_a(t)$.

3 Filter Analysis

In order to compare the filters, values of α and β have been chosen so as to have the same units (time), allowing a scale factor ϵ to be introduced so that

$$\beta = \epsilon\alpha$$

In accordance with Wiener filtering [12] and section 1.5.1, the filters were compared in the frequency domain. The frequency domain representations of the two filters are given by:

$$\exp[\frac{-|t|}{\alpha}] \leftrightarrow \frac{1}{\alpha\pi}\frac{1}{\frac{1}{\alpha^2}+\omega^2} \quad \text{and} \quad \exp[-\frac{t^2}{\beta^2}] \leftrightarrow \frac{1}{\beta^2}\sqrt{\frac{\pi}{2}}\exp[-\frac{\omega^2\beta^2}{4}]$$

where $\leftrightarrow$ denotes 'is the Fourier transform of'.

These are then normalised to have unit gain at dc, giving:

$$\hat{f}_b(\omega) = \frac{1}{\alpha^2}\frac{1}{\frac{1}{\alpha^2}+\omega^2} \quad \text{and} \quad \hat{g}_b(\omega) = \exp[-\frac{\omega^2\beta^2}{4}]$$

In order to find α and β such that for fixed bandwidth $B_{\alpha,\beta}$ both filters would contain the same proportion (k) of their total energy in the frequency domain the following equation was solved

$$\frac{\int_0^{B_{\alpha,\beta}}(\exp[-\frac{\omega^2\beta^2}{4}])^2 d\omega}{\int_0^{\infty}(\exp[-\frac{\omega^2\beta^2}{4}])^2 d\omega} = k = \frac{\int_0^{B_{\alpha,\beta}}(\frac{1}{\alpha^2}\frac{1}{\frac{1}{\alpha^2}+\omega^2})^2 d\omega}{\int_0^{\infty}(\frac{1}{\alpha^2}\frac{1}{\frac{1}{\alpha^2}+\omega^2})^2 d\omega}$$

Setting $B_{\alpha,\beta} = 1$ and solving simultaneously for α and β gives for k=0.9, ϵ=1.385. The results for k=0.9 are plotted in figure 4 for filters in the time domain and for their normalised frequency representation in figure 5. Note that the dotted line represents the exponential and the continuous line the Gaussian.

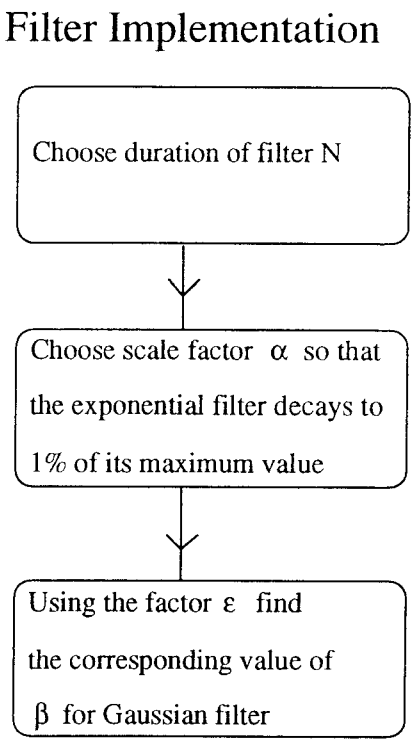


Figure 2: Method for Implementing Filters

4 Experiments and Filter Implementation

There follows a description of how the two filters were implemented and compared experimentally, the results appear in the next section (section5). Both filters were implemented as FIR filters in the time domain. To minimise the error due to the FIR implementation a low cut off (around 1%) was used. The value of $\epsilon = 1.385$ (see section 3), and the value $\alpha \approx 2$ was used (see figure 2).

The experiments given here are outlined in the Introduction. They break down into two categories: filter performance (for partial detection) and onset detection.

4.1 Smoothing For Partial Detection

For two pieces of music Schubert piano trio in E♭ Major (first 5 seconds), and the 1st Brandenburg Concerto by Bach (first 5 seconds), varying levels of noise were added to the original signal giving signal to noise ratios of 10dB, 20dB, 30dB and 40dB. Let ρ denote the finite signal to noise ratio (in dB) of the signal $x(t)$ which transforms under the MFT to

$$\hat{x}^{\rho}(i,j,n) \quad \text{where} \quad \rho = 10\log(\frac{\sigma^2_{signal}}{\sigma^2_{noise}})$$

σ^2_{signal} and σ^2_{noise} are the variances of the signal and the added noise respectively.

4.1.1 Filter Performance

Before being filtered, the transformed signal is phase-differenced in time according to the partial model (see section 1.5.2 and Pearson [13]).

$$\angle\hat{z}(i,j,n) = \angle(\hat{x}(i,j,n)\hat{x}^*(i-1,j,n))$$

and

$$\mid \hat{z}(i,j,n) \mid = \sqrt{\mid \hat{x}(i,j,n) \mid\mid \hat{x}(i-1,j,n) \mid}$$

where $\hat{z}(i,j,n)$ is the time phase-differenced MFT coefficient.

The filters (bilateral exponential and Gaussian equivalent) were each applied to the transformed data for $\rho \in \{10,20,30,40\}$. Then for each filtered noisy sample, the residual noise $e_{\hat{y}^\rho(i,j,n)}$ is calculated by subtracting the unfiltered clean MFT coefficients from the filtered noisy ones:

$$e_{\hat{y}^\rho(i,j,n)} = \hat{y}^\rho(i,j,n) - \hat{z}(i,j,n)$$

where $\hat{y}^\rho(i,j,n)$ are the filtered time phase-differenced MFT coefficients of the signal $x(t)$ with ρdB of noise added and $\hat{z}(i,j,n)$ time phase-differenced MFT coefficient of the same signal $x(t)$ with no noise added, where

$$\hat{y}^\rho(i,j,n) = \sum_{l,m} h(j-l)f(i-m)\hat{z}^\rho(i,j,n)$$

where h(j) and f(i) are as defined in section 2.1.

The residual noise is the amount of sound signal that does not conform to the partial model (see section 1.5.2). The filter can be considered as a 'partial filter' since it uses the partial model (see section 1.5.2 and [13]) to smooth the transformed signal $x(t)$. The coefficient $\hat{y}^\rho(i,j,n)$ is smoothed and phase-differenced in time so the difference between it and the unsmoothed clean MFT coefficient $\hat{z}(i,j,n)$ is proportional to the amount of $\hat{y}^\rho(i,j,n)$ that does not conform to the partial model: in other words the amount of noise $n(t)$. For each of the above filters and each of the sound signals (Schubert and Bach), the noise residual energy to clean signal energy is graphed. The signal to residual noise energy ratio (SRNR) defined as:

$$\rho_e = \sum_{i,j} \frac{\mid \hat{z}(i,j,n) \mid^2}{\mid e_{\hat{y}^\rho(i,j,n)} \mid^2}$$

with $\hat{z}(i,j,n)$ and $e_{\hat{y}^\rho(i,j,n)}$ defined as above.

The lower the SRNR is then the more signal has been detected by smoothing with the filter. As the input noise ρ increases so should the SRNR. The main comparison is between the different filters in order to find a difference between their different SRNR values.

4.1.2 Filter Length

This experiment is similar to the one described in section 4.1.1 but was performed for only one musical piece (Schubert). The exponential factor (α) is varied for different values of ρ to find the SRNR as a function of α.

4.2 Onset Detectors

In sections 4.2.1 and 4.2.2, two different methods for onset detection are discussed, both relying on filters used in the partial detection processes (see section 4.1). They would therefore be easier to implement than that used by Pearson [13] which was the time differenced version of his partial detection window. By simply reprocessing the partial detection filter output another pass through the MFT coefficients would be avoided. In section 4.2.2 the output of the anticausal filter is divided by that of the causal filter and in section 4.2.1 the output of the partial detection filter is differenced in time.

4.2.1 Time Differencing Filtered Output

Taking the smoothed, phase-differenced transformed coefficients $\hat{y}(i,j,n)$, and differencing there magnitudes according to

$$\frac{\partial \hat{z}(i,j,n)}{\partial t} \approx \frac{\mid \hat{z}(i,j,n) \mid - \mid \hat{z}(i-1,j,n) \mid}{\Xi(n)} \quad i \geq 0$$

where $\Xi(n)$ is temporal sampling interval and the difference equation is a first order approximation to the differential of $\hat{z}(i,j,n)$ at the point (i,j) on the time frequency plane. Since onsets are being specifically considered and not offsets, the differential is set to zero for negative values, giving the modified differential

$$\frac{\partial^{+} \hat{z}(i,j,n)}{\partial t} \approx \begin{cases} \frac{|\hat{z}(i,j,n)|-|\hat{z}(i-1,j,n)|}{\Xi(n)} & \text{for } \mid \hat{z}(i,j,n) \mid - \mid \hat{z}(i-1,j,n) \mid \geq 0 \\ 0 & \text{else} \end{cases}$$

which is non zero for onsets only.

4.2.2 The Quotient Onset Detector

The input signal is transformed (MFT) and phase-differenced as described in section 4.1.1. The filters are applied in time, one forwards and one backwards (see section 2.1). When the anti-causal filter first detects a partial the causal filter will still be filtering noise see figure 3. The quotient of these two filters $\frac{f_a(t)}{f_c(t)}$ will be largest at the onset of a partial, when $f_a(t)$ is large and $f_c(t)$ is small.

While both the causal and anti-causal filters are filtering a partial their magnitude should be roughly equal giving a magnitude of unity. To avoid division by zero, the filters overlap by one sample.

5 Results and Conclusions

In this section, the results from the previous section's experiments are given, along with conclusions drawn from them.

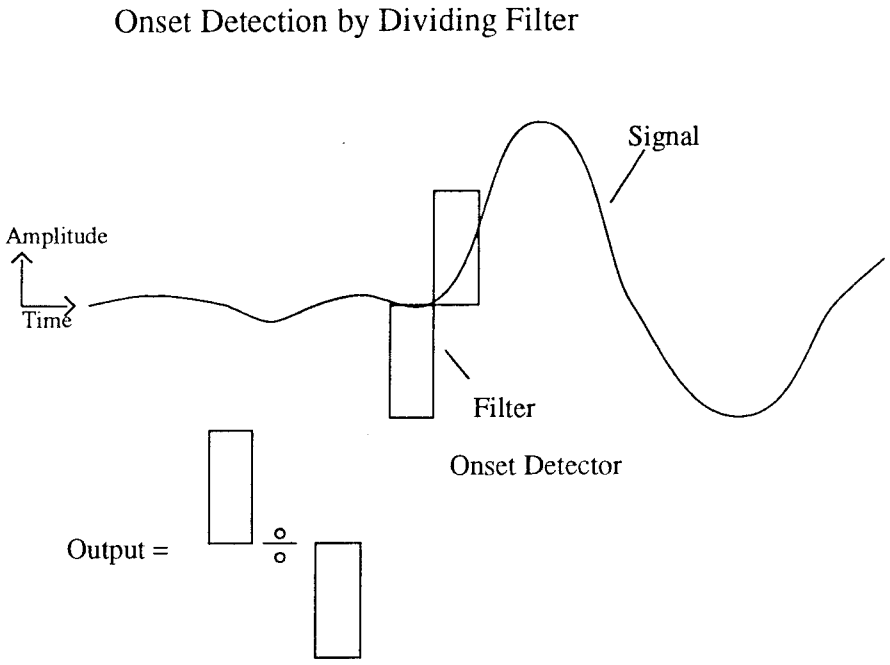


Figure 3: Onset Detection By Dividing Anti-Causal Filter by Causal Filter

5.1 Results For Filter in Noise Comparison

This experiment is described in section 4.1.1. The graphs of SRNR (y axis) to input signal SNR (ρ) in dB (x axis) are given in figures 8 and 9 for the Schubert Piece and figures 10, 11 for Bach's Woodwind Trio and $\alpha = 1.8$.

The bilateral exponential filter is compared with its Gaussian equivalent for all cases and for MFT levels 11 and 13. It is seen that for all cases the exponential filter performs better than the Gaussian one, though the difference is not great. It can be concluded that compared with the Gaussian filter the exponential filter is better suited for audio restoration as it performs at least as good as the Gaussian and is quicker and more accurate when implemented recursively.

5.2 Results For Scale Choice

These were plotted (see figures 6 and 7) for the anti-causal exponential filter applied to the Schubert Piece for various values of α and various values of Gaussian noise (added to the original signal).

The x axis of each graph corresponds to α with data points at $\alpha \in \{0.52, 1.02, 2.04, 4.08, 8.16\}$. The y axis corresponds to the gain in SRNR (see section 5.1 and section 4.1.1), measured in dB. The title of each graph is the amount of noise added to the original signal (ρ). All results were for MFT level 11.

As is expected the gain of the filter increases with α. Note how as the input signal SNR

increases the SRNR gain increases. For less noisy input signals the smaller filter performs better than when there is more noise, and the larger filter performs better for more noise. The change in relative gain flattens out at around $\alpha \approx 2.5$.

5.3 Results For Onset Detection By Time Differencing

This experiment was as described in section 4.2.1. Level 11 of the MFT was used because it has a reasonably high time resolution of 11.2 mS, required for detection of onset times.

The differencer was implemented on the output of two filters $\exp[\frac{-|t|}{\alpha}]$ and $\exp[\frac{-t}{\alpha}]$. The results can be seen in figures 16 and 17 next to the MFT (level 11) for both Schubert and Bach. These figures have time along the x axis, frequency along the y axis (3 kHz maximum) and the intensity denotes the amplitude of the signal

It can be seen that this method is much more suitable for onset detection, and its ease of implementation (figure 4.2.2) makes it suitable for use in audio restoration.

5.4 Results For Quotient Onset Detector

This experiment is as described in section 4.2.2. Again using MFT level 11 as discussed in section 5.3.

In figures 12, 13, 14 and 15, results of the onset detection are given beside the MFT of the appropriate piece of music (Schubert's Piano Trio in E ♭ major and the start of Bach's 1st Brandenburg Concerto). These figures are similar to those described in section 5.3. The output is on a log scale, the threshold being chosen empirically.

Figure 12 is for the anti-causal filter output divided by the causal filter output.

$$d_1(i,j,n) = \frac{f_c(i,j,n)}{f_a(i,j,n)}$$

This is the most basic quotient filter. As can be seen (from the theory and the figures) these filters effectively normalise the energy of the entire signal, which should mean that, for example, the onset of the i^{th} $(i \in N^+)$ partial of a note would be just as obvious as the onset of the fundamental. The drawback of this as can be plainly seen from the figures is that the slightest blip, can be construed as an onset. Normally an onset detector would work in conjunction with a partial detector and so 'false starts' would be ignored. The quality of the quotient detector is too poor to use in a noisy environment.

Figure 13 is for the anti-causal filter output divided by the sum of the causal and ant-causal filters' output.

$$d_2(i,j,n) = \frac{f_a(i,j,n)}{f_a(i,j,n) + f_c(i,j,n)}$$

Figure 14 is for the anti-causal filter's output divided by the magnitude of the causal filter's output

$$d_3(i,j,n) = \frac{f_a(i,j,n)}{\mid f_c(i,j,n) \mid}$$

Figure 15 is for the anti-causal filter's output divided by the magnitude of the sum of the causal and the anti-causal filters' output.

$$d_4(i,j,n) = \frac{f_a(i,j,n)}{\mid f_a(i,j,n) + f_c(i,j,n) \mid}$$

The magnitude of the denominator filter is used in figures 14 and 15. These filters are looking for a change in magnitude at the onsets, independently of the partial model (which requires neighbouring bins to have the same phase) by ignoring the phase.

By dividing the anti-causal filter by itself and the causal filter (see figures 13 and 15) the change in output of the quotient filter is less dramatic at onsets than if the denominator was the causal filter alone.

It can be concluded that even though this method of onset detection is very easily implementable and reduces the amount of processing (see sections 4.2.2 and 4.2.1) it is far too susceptible to rapid changes due to background noise, and therefore is not suitable for use in audio restoration.

5.5 Summary Of Conclusions

For filter performance the exponential filter and Gaussian filter were very alike. The exponential filter is easier and more accurate if implemented recursively, computationally cheaper and is, therefore, better suited to audio restoration than the Gaussian filter.

The best filter length is one of $\alpha \approx 2$. This is where the filter performance curves flatten out.

The quotient onset detector is not suitable for onset detection due to its sensitivity to rapid changes in the background, whereas the time differencing method is less sensitive and therefore better suited to audio restoration.

6 Further Work

Using the filter described, it is proposed to segment two recordings of the same musical piece. Recording A is a noisy recording of the piece requiring noise removal and recording B is clean, containing only harmonic sounds and possibly recorded digitally. A correspondence function would be applied matching features pertaining to the same musical events in both recordings. This would take into consideration the possibility of a partial missing or an extraneous one being present for example. Using the information of correspondence recording B would be warped (in time and frequency) to match exactly recording A. This warped version of recording B would then be used as a mask to filter the noise from recording A giving recording A', a restored version of recording A.

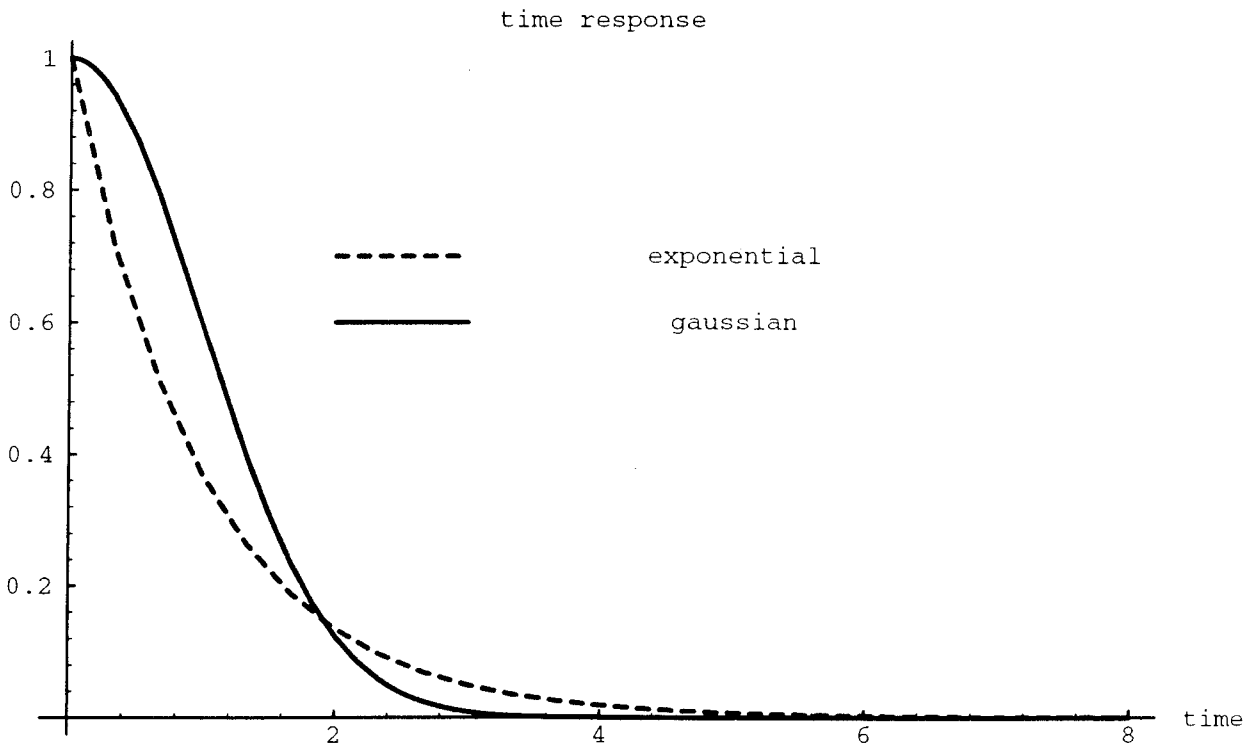


Figure 4: Gaussian and exponential filters in time $\alpha = 1, \beta = 1.385$

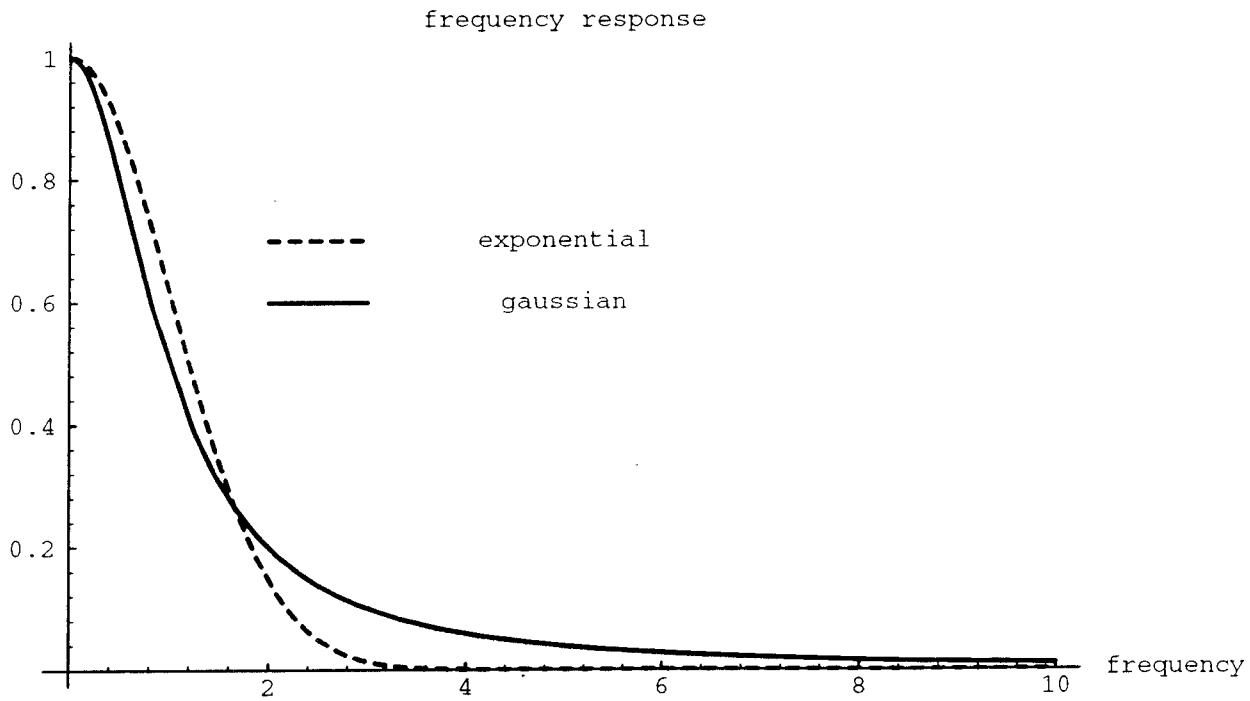
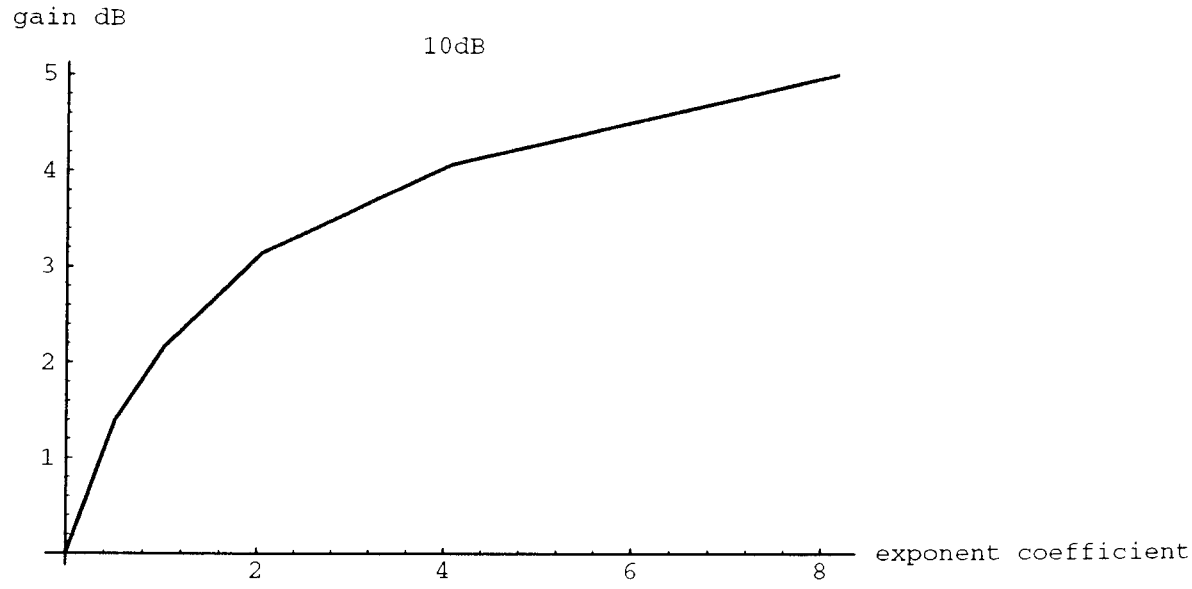


Figure 5: Gaussian and exponential filters in frequency$\alpha = 1, \beta = 1.385$



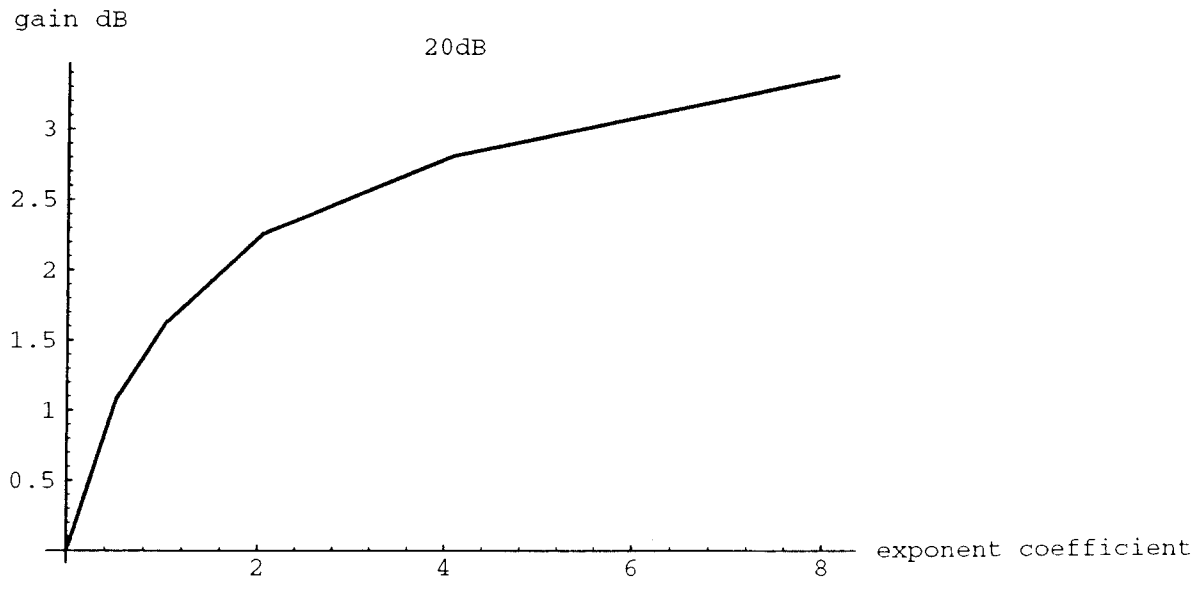
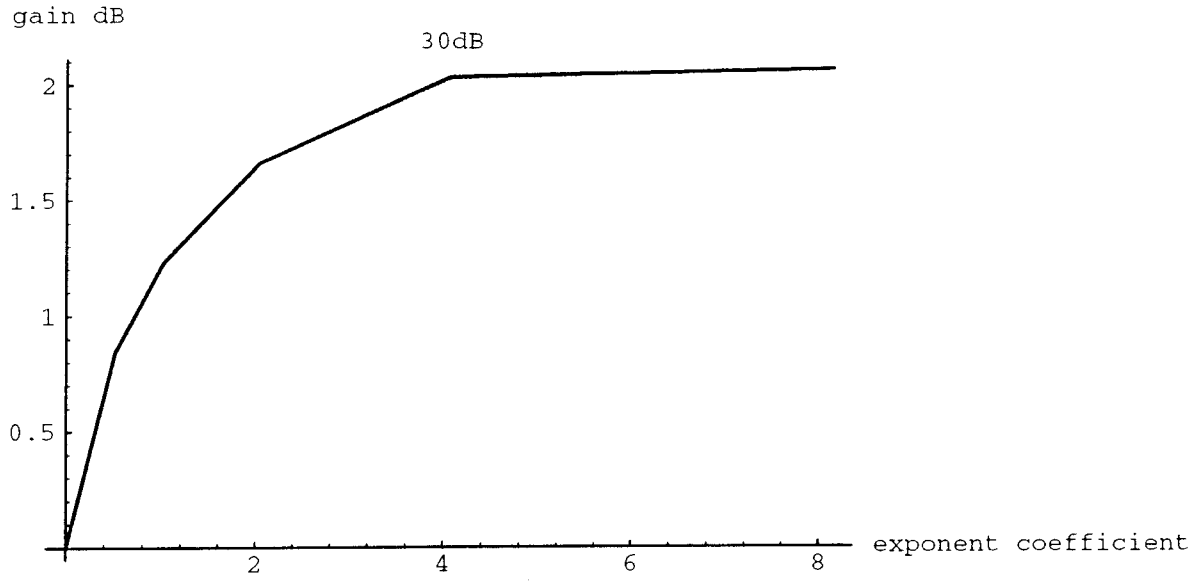


Figure 6: Signal to noise residue gain versus linear exponential factor α



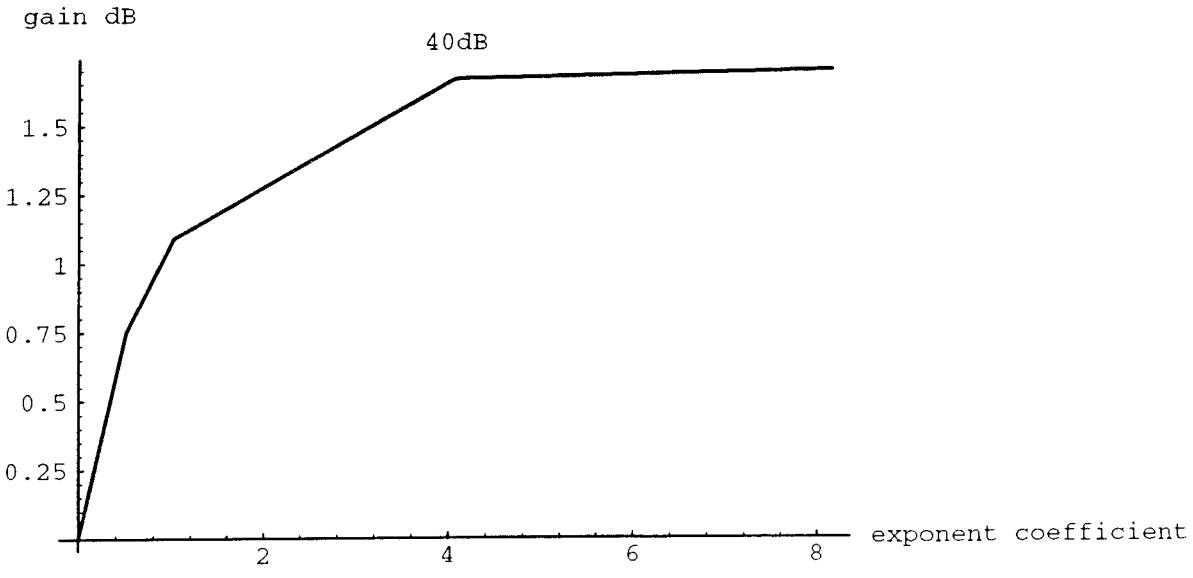


Figure 7: Signal to noise residue gain versus linear exponential factor α

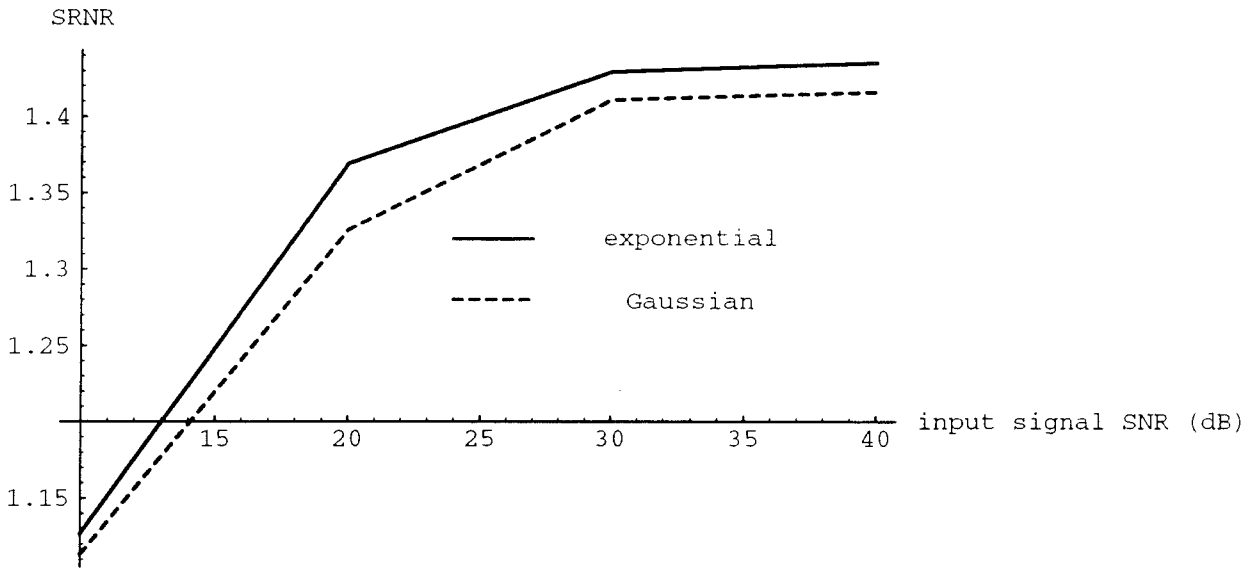


Figure 8: Filter performance for Schubert at MFT level11

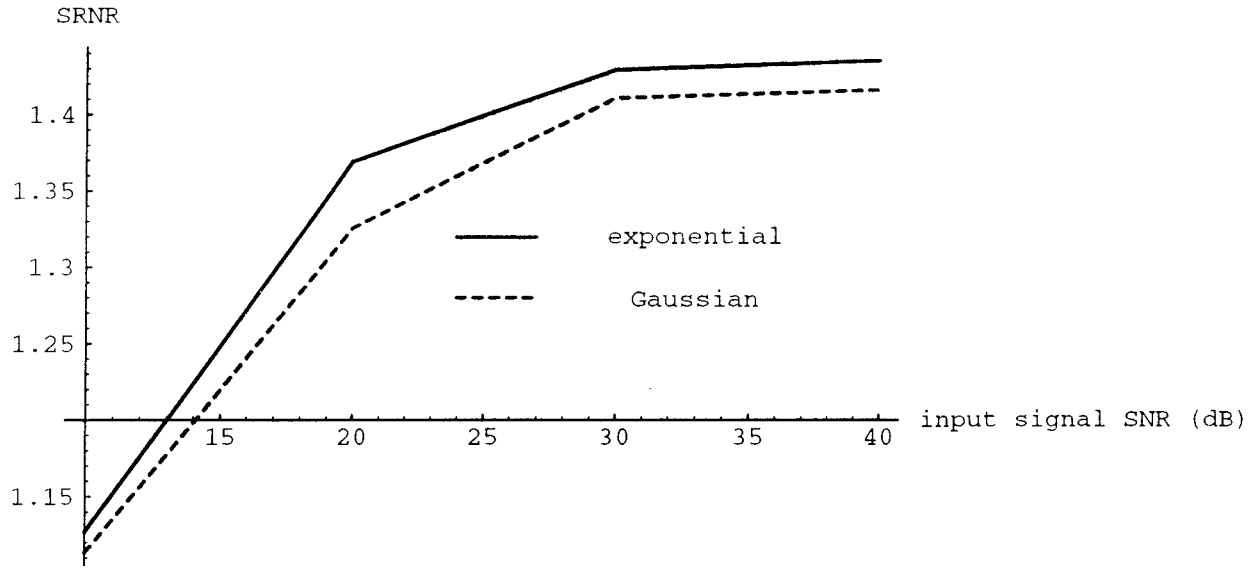


Figure 9: Filter performance for Schubert at MFT level13

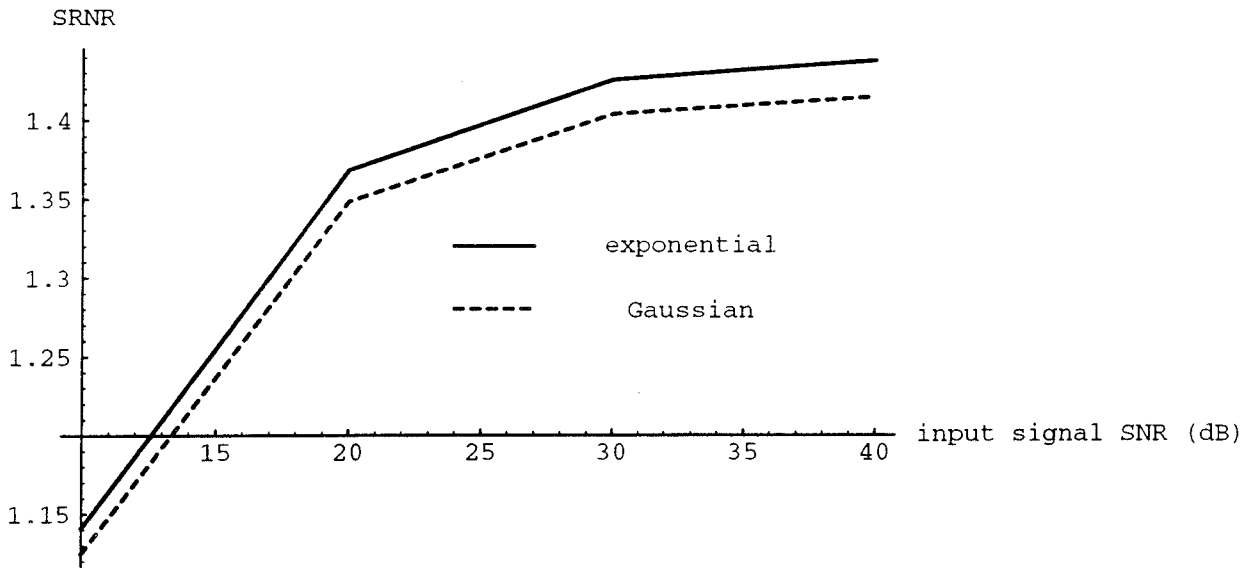
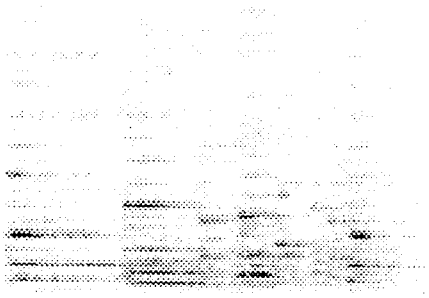
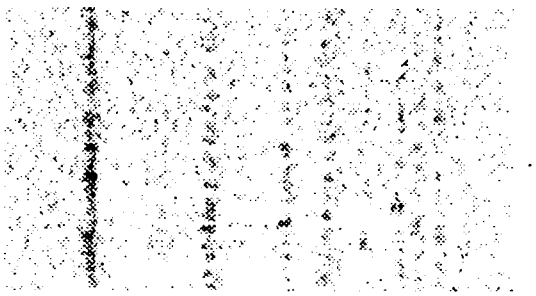


Figure 11: Filter performance for Bach at MFT level11

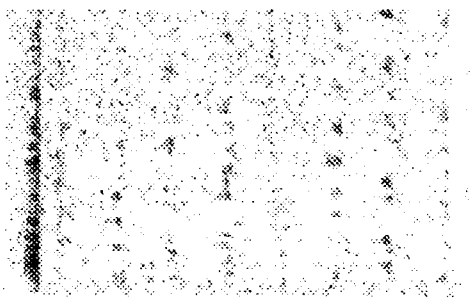


a.Schubert MFT level 11

b.Bach MFT level 11



c.Quotient onset detection Schubert -70dB

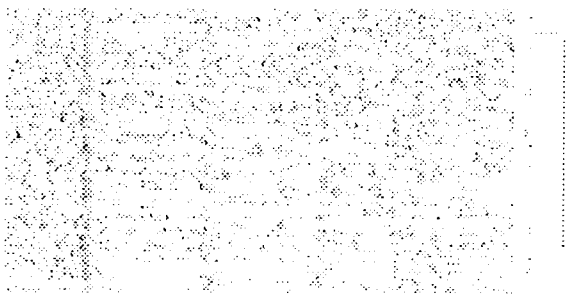


d.Quotient onset detection Bach -50dB

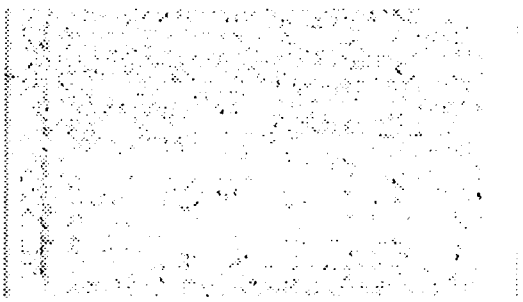
Figure 12: Onset detection of form f_a divided by f_c

a.Schubert MFT level 11

b.Bach MFT level 11



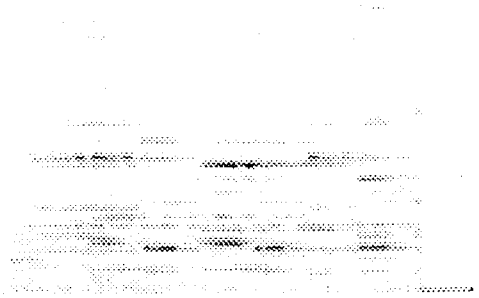
c.Quotient onset detection Schubert -50dB



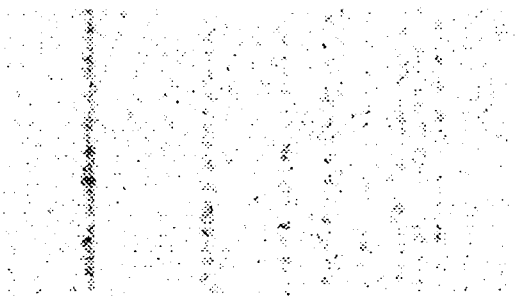
d.Quotient onset detection Bach -40dB

Figure 13: Onset detection of form f_a divided by $f_c + f_a$

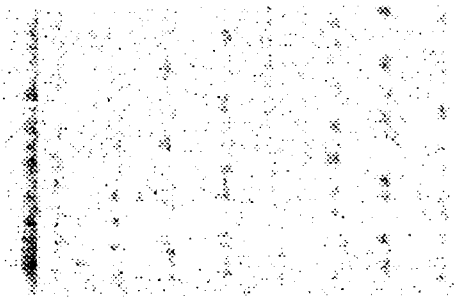
a.Schubert MFT level 11



b.Bach MFT level 11

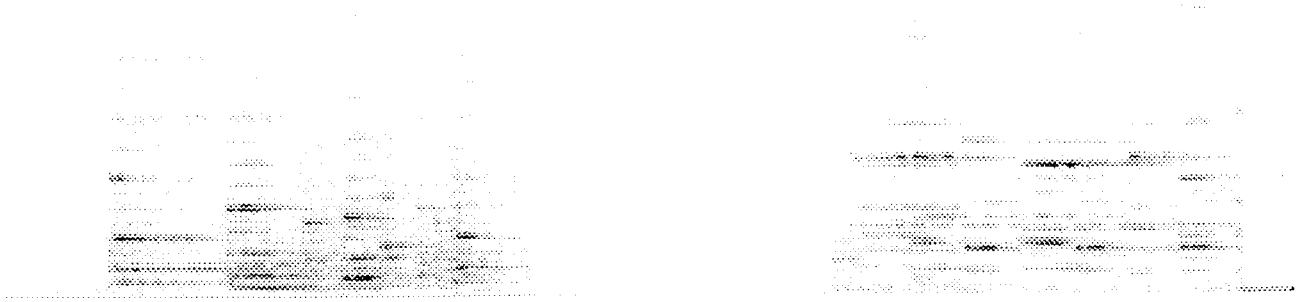


c.Quotient onset detection Schubert -70dB



d.Quotient onset detection Bach -60dB

Figure 14: Onset detection of form f_a divided by $|f_c|$



a.Schubert MFT level 11

b.Bach MFT level 11

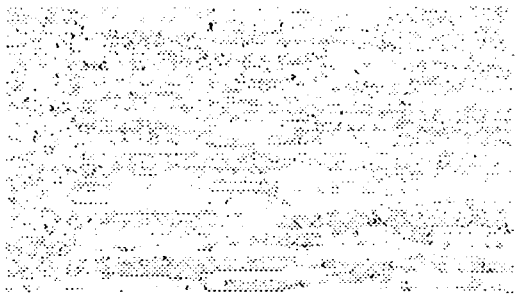
c.Quotient onset detection Schubert -70dB

d.Quotient onset detection Bach -60dB

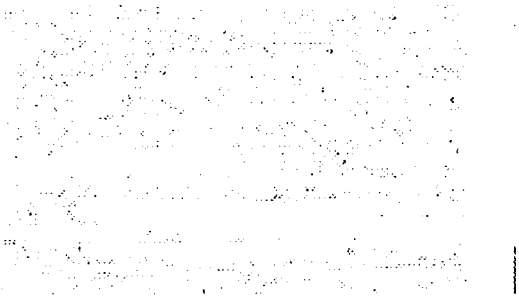
Figure 14: Onset detection of form f_a divided by $|f_c|$

a.Schubert MFT level 11

b.Bach MFT level 11



c.Quotient onset detection Schubert -35dB

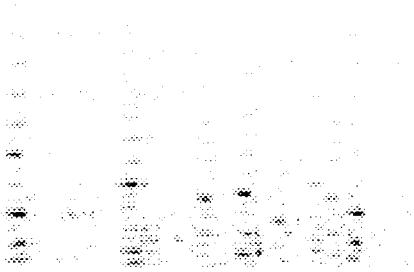


d.Quotient onset detection Bach -40dB

Figure 15: Onset detection of form f_a divided by $|f_c + f_a|$

a.MFT level 11

b.Filtered (f_a)



c.Filtered (f_b)

Figure 16: Onset detection time differencing for Schubert

a.MFT level 11

b.Filtered (f_b)

Figure 17: Onset detection time differencing for Bach

References

[1] Andrew Calway. *The Multiresolution Fourier Transform: A general Purpose Tool for Image Analysis.* PhD thesis, Department of Computer Science, The University of Warwick, UK, September 1989.

[2] Chris Chafe. Source separation and note identification in polyphonic music. Technical Report STAN-M-34, Stanford University, Department of Music, April 1986.

[3] Chris Chafe, David Jaffe, Kyle Kashima, Bernard Mont-Reynaud, and Julius Smith. Techniques for note identification in polyphonic music. In *Proceedings International Computer Music Conference*, 1985.

[4] Ingrid Daubechies. Orthonormal bases of compactly supported wavelets. *Communications on Pure and Applied Mathematics*, XLI:909–996, 1988.

[5] Martin Piszczalski et al. Performed music: Analysis, synthesis, and display by computer. *Journal of the Audio Engineering Society*, 29:38–46, 1981.

[6] D. Gabor. Acoustical quanta and the theory of hearing. *Nature*, 1947.

[7] Smith George. Method for analysis/synthesis. *Audio Engineering Society*, 1992.

[8] Rudolf Rasch Joos Vos. The perceptual onset of musical tones.

[9] Venkatsen Mathews. *A TextBook Of Quantum Mechanics.* Tata McGraw Hill, 1975.

[10] B Mont-Reynaud. The bounded-q approach to time-varying spectral analysis. Technical Report STAN-M-28, Stanford University, Department of Music.

[11] James A. Moorer. *On the Segmentation and Analysis of Continuous Musical Sound by Digital Computer.* PhD thesis, Stanford University, Department of Music, 1975.

[12] Athanasios Papoulis. *Signal Analysis.* McGraw-Hill, 1977.

[13] Edward R.S. Pearson. *The Multiresolution Fourier Transform and its apllication to Polyphonic Audio Analysis.* PhD thesis, Warwick University, 1991.

[14] Edward R.S. Pearson and R. G. Wilson. Musical event detection from audio signals within a multiresolution framework. In *Proceedings International Computer Music Conference*, 1990.

[15] L. R. Rabiner and B. Gold. *The Theory and Application of Digital Signal Processing.* Prentice-Hall, 1975.

[16] Curtis Roads. Granular synthesis of sound. In Curtis Roads and John Strawn, editors, *Foundations of Computer Music.* MIT Press, 1985.

[17] E.R.S Pearson Roland G Wilson, A.D Calway and A.R Davies. An introduction to the multiresolution fourier transform and its applications. Technical Report RR204, Department of Computer Science, University of Warwick, 1992.

[18] Xavier Serra. A system for sound analysis/transformation/synthesis based on a deterministic plus stochastic decomposition. Technical Report STAN-M-58, Stanford University, CCRMA, Department of Music, 1989.

[19] Julius O. Smith and Xavier Serra. Parshl: An analysis/synthesis program for non-harmonic sounds based on a sinusoidal representation. In *Proceedings International Computer Music Conference*, 1986.

[20] Charles Watson. *The Computer Analysis of Polyphonic Music.* PhD thesis, The University of Syndney, Australia, 1986.

[21] Roland G. Wilson, Andrew D. Calway, and Edward R.S. Pearson. A generalised wavelet transform for fourier analysis: the multiresolution fourier transform and its application to image and audio signal analysis. *IEEE Transactions on Information Theory*, 1992.

[22] Roland G. Wilson and M. Spann. Finite prolate spheroidal sequences and their applications I-II. *IEEE Transactions on Pattern Recognition and Machine Intelligence*, 1987.

[23] Roland G. Wilson and M. Spann. *The Uncertainty Principle in Image Processing.* Research Studies Press, 1988.